# ENCRYPTION USING FPGA

NOR ROBAINI BINTI IBRAHIM

This thesis is submitted as partial fulfillment of the requirements for the award of the Bachelor of Electrical Engineering (Hons.) (Electronics)

Faculty of Electrical & Electronics Engineering
Universiti Malaysia Pahang

MEI, 2008

## ACKNOWLEDGEMENT

Assalamualaikum warahmatullahi wabarahkatu,

Thank you Allah for give me this opportunity to finish my undergraduate project. It is a very great pleasure for me to acknowledge the contribution of a large number of individuals that being supportive throughout this year. First of all, I would like to thank my supervisors, Puan Nurul Hazlina binti Noordin, for provide me precious helps, supports and motivation throughout the development of this project.

I would like to dedicate appreciation to my friends and course mates. Not to forget my beloved roommates who always lend me shoulders to cry on. We have gone through thick and thin together. All the courage and valuable memory you gave will never be forgotten. Thank you for always being by my side.

I also would like to acknowledge my parents and siblings. Thank you for your support and motivation. I am gratefully acknowledged the support, encouragement, and patience of my families. I am very happy to have family that always loves me and care about me. Without them, I will not be able to finish this project. Last but not least to all other peoples whose are not mention here. Your contributions are very much appreciated.

Thank you very much.

# ABSTRACT

Advanced Encryption Standard (AES), a Federal Information Processing Standard (FIPS), is an approved cryptographic algorithm that can be used to protect electronic data. The AES algorithm is a block cipher that can encrypt and decrypt digital information. The AES algorithm is capable of using cryptographic keys of 128, 192, and 256 bits, this project implements the 128 bit standard on a Field Programming Gate Array (FPGA) using the VHDL, a hardware description language.

# ABSTRAK

Piawaian Penyulitan Maju (AES), satu Federal Information Processing Standard (FIPS), adalah diluluskan di mana kriptografik algoritma yang boleh digunakan untuk melindungi data elektronik. Algoritma AES adalah satu sifar blok yang boleh encrypt dan decrypt maklumat digital. Algoritma AES adalah berupaya menggunakan cryptographic kunci-kunci 128, 192, dan 256 bits, di mana projek ini menggunakan 128 bit mata piawaian pada Field Programming Gate Array (FPGA) menggunakan VHDL, satu perkakasan bahasa penggambaran.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

## LIST OF ABBREVIATIONS

| | |
|---|---|
| AES | Advanced Encryption Standard |
| Affine | A transformation consisting of multiplication by a matrix followed by Transformation the addition of a vector. |
| Array | An enumerated collection of identical entities (e.g., an array of bytes). |
| Bit | A binary digit having a value of 0 or 1. |
| Block | Sequence of binary bits that comprise the input and output. The length of a sequence is the number of bits it contains. Blocks are also interpreted as arrays of bytes. |
| Byte | A group of eight bits that is treated either as a single entity or as an array of 8 individual bits. |
| Cipher | Series of transformations that coverts plaintext to ciphertext using the Cipher Key. |
| Cipher Key | Secret, cryptographic key that is used by the Key Expansion routine to generate a set of Round Keys; can be pictured as a rectangular array of bytes, having four rows and n columns. |
| Ciphertext | Data output from the Cipher or input to the Inverse Cipher. |
| Inverse Cipher | Series of transformations that converts ciphertext to plaintext using the Cipher Key. |
| Key Expansion | Routine used to generate a series of Round Keys from the Cipher Key. |
| Plaintext | Data input to the Cipher or output from the Inverse Cipher. |
| Rijndael | Cryptographic algorithm specified in this Advanced Encryption Standard (AES). |

| | |
|---|---|
| Round Key | Round keys are values derived from the Cipher Key using the Key Expansion routine; they are applied to the State in the Cipher and Inverse Cipher. |
| State Intermediate | Cipher result that can be pictured as a rectangle array of bytes, having four rows and m columns. |
| S-box | Non-linear substitution table used in several byte substitution transformations and in the Key Expansion routine to perform a one-for-one substitution of a byte value. |
| Word | A group of 32 bits that is treated either as a single entity or as an array of 4 bytes. |
| Word | A group of 32 bits that is treated either as a single entity or as an array of 4 bytes. |

# CHAPTER 1

# INTRODUCTION

## 1.1 Overview

AES is an algorithm for performing encryption (and the reverse, decryption) which is a series of well-defined steps that can be followed as a procedure. The original information is known as plaintext, and the encrypted form as cipher text. The cipher text message contains all the information of the plaintext message, but is not in a format that can read by a human or computer without the proper mechanism to decrypt it; it should be resemble random gibberish to those not intended to read it. The encryption procedure is varied depending on the key which changes the detailed operation of the algorithm. Without the key, the cipher cannot be used to encrypt or decrypt. In the past, cryptography helped ensure secrecy in important communications, such as those of government covert operations, military leaders and diplomats. Cryptography has come to be in widespread use by many civilians who cannot have extraordinary needs for secrecy, although typically it is transparently built into the infrastructure for computing and telecommunications [1].

## 1.2 Objectives

This project is more towards on implement encryption technology with using FPGA as a device for encrypting data.

The objectives of this project are:

i. To study the encryption technology that widely use in communication as a guidance for secrecy.
ii. To implement Advanced Encryption Standard (AES) with using digital concept.

## 1.3 Problem Statement

There are four algorithms in AES which is subbytes transformation, shiftrows transformation, mixcolumns transformation and addroundkey transformation. Each of these algorithms has mathematical equation and it can be proved with using digital concept.

i. Subbytes transformation

$$b_i' = b_i \text{ xor } b_{(i+4)mod8} \text{ xor } b_{(i+5)mod8} \text{ xor } b_{(i+6)mod8} \text{ xor } b_{(i+7)mod8} \quad (1.1)$$

ii. Shiftrows transformation

$$s'_{r,c} = s_{r,(c+shift(r,Nb))modNb} \qquad \text{for } 0 < r < 4 \text{ and } 0 \leq c < Nb \quad (1.2)$$

iii. Mixcolumns transformation

$$a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\} \quad (1.3)$$

iv. Addroundkey transformation

$$[s'_{0,c}, s'_{1,c}, s'_{2,c}, s'_{3,c}] = [s_{0,c}, s_{1,c}, s_{2,c}, s_{3,c}] \text{ xor } [w_{round*Nb+c}] \quad (1.4)$$

## 1.4 Scope of Project

The scope of this project is the sum total of a project's products and their requirements or features. Sometimes the term scope is used to mean the totality of work needed to complete a project. The scope of this project is to implement encryption onto digital concept. Implement here means that execute the encryption system with using digital system. The type of standard that used in this project is Advanced Encryption Standard (AES) – Rijndael.

# CHAPTER 2

# LITERATURE REVIEW

## 2.1 Encryption

Cryptography or cryptology come from Greek word means that hidden and the verb is write or to speak (Wikipedia). Nowadays, cryptography widely used in daily life; examples includes the security of ATM cards, computer password, and electric commerce, that all depend on cryptography. Until modern times, cryptography almost referred to encryption, a process of transforming information known as plaintext using an algorithm to make it unreadable by human or computer without proper mechanism to decrypt it. Decrypt was the reverse operation of encryption.

There were two types of encryption; symmetric (private-key cryptography) and asymmetric (public-key cryptography). Symmetric-key cryptography refers to encryption method where both the sender and the receiver share the same key (refer Figure 2.1). There were several types of algorithm under symmetric-key cryptography such as Data Encryption Standard (DES), Triple DES (3DES), Twofish, Advanced Encryption Standard (AES), and many more. In asymmetric cryptography (public-key cryptography), users have a pair of cryptography key; public and private keys to encrypt or decrypt data (refer Figure 2.2). Rivest, Shamir, Adleman (RSA), Diffie-Hellman (& Merkle), Pretty Good Privacy (PGP), and Elliptical Curve Cryptography were some types of algorithm under public-key cryptography.

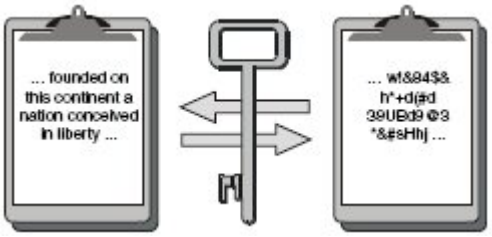


**Figure 2.1:** Encrypting and then decrypting with the same key

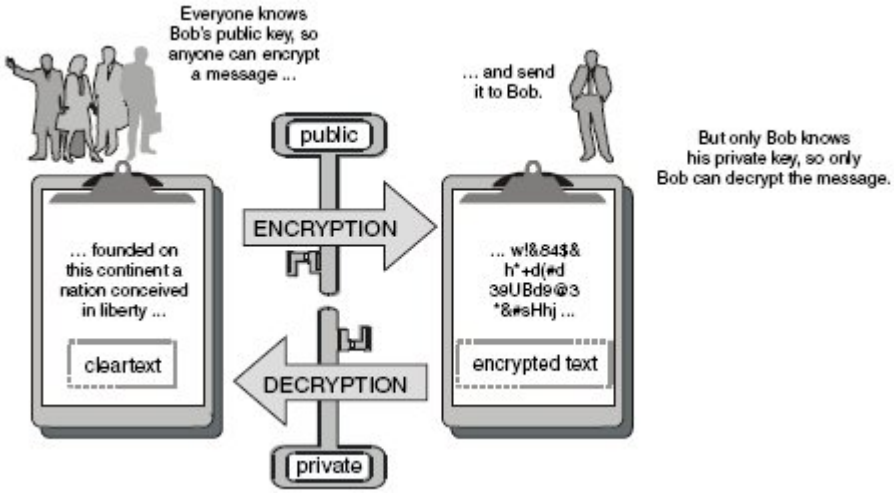


**Figure 2.2:** Encrypting and then decrypting with public-private key

### 2.1.1 Symmetric cryptography (Private-key cryptography)

Symmetric cryptography or algorithms are a part of encryption technology that use slightly related, often identical, cryptographyc keys for both encryption and decryption.

The encryption key is slightly related to the decryption key, in that they may be identical or there is a simple transform to go between the two keys. The keys represent a shared secret between two or more parties that can be used to maintain private information (Figure 2.1).

Other terms for symmetric encryption are secret-key, single key, shared-key, one-key and eventually private-key encryption [2]

#### 2.1.1.1 Data Encryption Standard (DES)

The Data Encryption Standard (DES) is a cipher (a method for encrypting information) selected as an official Federal Information Processing Standard (FIPS) for the United States in 1976 [3]. The algorithm was initially controversial, with classified design elements, a relatively short key length, and suspicions about a National Security Agency (NSA) backdoor. DES consequently came under intense academic study, and motivated the modern understanding of block ciphers and their cryptanalysis.

DES is now considered to be insecure for many applications. This is chiefly due to the 56-bit key size being too small; DES keys have been broken in less than 24 hours. There are also some analytical results which demonstrate theoretical weaknesses in the cipher, although they are infeasible to mount in practice. The algorithm is believed to be practically secure in the form of Triple DES, although there are theoretical attacks. In recent years, the cipher has been superseded by the Advanced Encryption Standard (AES).

#### 2.1.1.2 Advanced Encryption Standard (AES)

In cryptography, the Advanced Encryption Standard (AES), also known as Rijndael, is a block cipher adopted as an encryption standard by the U.S. government. AES was announced by National Institute of Standard and Technology (NIST) as U.S. FIPS PUB 197 (FIPS 197) on November 26, 2001 after a 5-year standardization process. It became effective as a standard on May 26, 2002. As of 2006, AES is one of the most popular algorithms used in symmetric key cryptography [4].

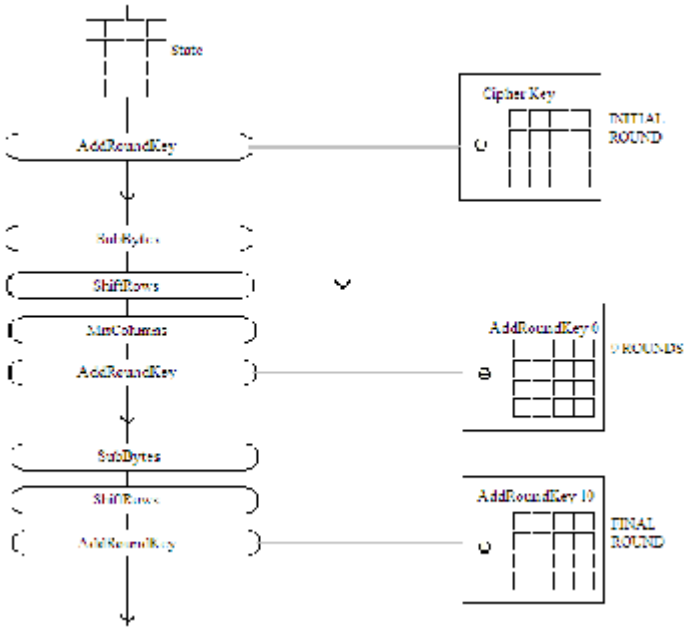


**Figure 2.3:** AES algorithms process

For the initial round, the state only goes through the addroundkey algorithm. Then, it go for the subbytes, shiftrows, mixcolumns, and addroundkey for 9 times before it goes to the final round that involved all the four algorithms except mixcolumns transformation. After goes through the entire algorithm, the state came out as ciphertext.

### 2.1.1.3 Twofish

In cryptography, Twofish is a symmetric key block cipher with a block size of 128 bits and key sizes up to 256 bits. It was one of the five finalists of the Advanced Encryption Standard contest, but was not selected for standardisation. Twofish is related to the earlier block cipher Blowfish.

Twofish's individual features are the use of pre-computed key-dependent S-boxes, and a relatively complex key schedule. One half of an n-bit key is used as the actual encryption key and the other half of the n-bit key is used to modify the encryption

algorithm (key-dependent S-boxes). Twofish borrows some elements from other designs; for example, the pseudo-Hadamard transform (PHT) from the SAFER family of ciphers. Twofish uses the same Feistel structure as DES. On most software platforms Twofish is slightly slower than Rijndael (the chosen algorithm for Advanced Encryption Standard) for 128-bit keys, but somewhat faster for 256-bit keys [5].

Twofish was designed by Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, and Niels Ferguson; the "extended Twofish team" who met to perform further cryptanalysis of Twofish and other AES contest entrants included Stefan Lucks, Tadayoshi Kohno, and Mike Stay.

The Twofish cipher has not been patented and the reference implementation is placed in the public domain, free to use for anyone

### 2.1.2 Asymmetric cryptography (Public-key cryptography)

Public-key cryptography, also known as asymmetric cryptography, is a form of cryptography in which a user has a pair of cryptographic keys; a public key and a private key [6]. The private key is kept secret, while the public key may be widely distributed. The keys are related mathematically, but the private key cannot be practically derived from the public key. A message encrypted with the public key can be decrypted only with the related private key (Figure 2.2).

The two main branches of public key cryptography are:

- **Public key encryption**: a message encrypted with a recipient's public key cannot be decrypted by anyone except the recipient possessing the corresponding private key. This is used to ensure confidentiality.
- **Digital signatures**: a message signed with a sender's private key can be verified by anyone who has access to the sender's public key, thereby proving that the sender signed it and that the message has not been tampered with. This is used to ensure authenticity.

#### 2.1.2.1 Pretty Good Privacy (PGP)

PGP Encryption (Pretty Good Privacy) is a computer program that provides cryptographic privacy and authentication. It was originally created by Philip Zimmermann in 1991 [7].

PGP encryption using public-key cryptography and includes a system that binds the public-keys to a user name. PGP approach in four ways:

- **Encryption/ Decryption**: When encrypting a message, the sender uses the public key half of the recipient's linked key pair to encrypt a symmetric cipher session key. That session key is used, in turn, to encrypt the plaintext of the message. The recipient of a PGP-encrypted message decrypts the session key using his private key (the session key was previously encrypted using his public key by the sender). Next, he decrypts the cyphertext of the message using the session key.

- **Digital signatures**: Compares this message digests with the message digest computed himself from the plaintext. If the signature matches the received

plaintext's message digest, it must be presumed that the message received has not been tampered with, either on purpose or accidentally.

- **Web of trust**: Users must verify some means that the public key in a certificate actually does belong to the person/entity claiming it. PGP products have included an internal certificate 'vetting scheme' to assist with this.

- **Certificate**: A trust signature indicates both that the key belongs to its claimed owner and that the owner of the key is trustworthy to sign other keys at one level below their own.
- **Security quality**: Cryptographic security of PGP encryption depends on the assumption that the algorithms used are unbreakable by direct cryptanalysis with current equipment and techniques.

## 2.2 Field Programming Gate Array (FPGA)

Field Programming Gate Array (FPGA) is a semiconductor device containing programmable logic components and programmable interconnects (Wikipedia). There were a lots of FPGA applications include digital signal processor (DSP), defense system, speech recognition, cryptography, bioinformatics, and growing range of other areas. FPGA offered any area of algorithm that can make use of the massive parallelism by their architecture. One such area is code breaking of cryptography algorithm.

### 2.2.1 Verilog Hardware Description Language (VHDL)

Verilog Hardware Description Language (VHDL) is commonly used as a design-entry-language for FPGA and application. Specific design integrated circuits in electronic design automation of digital circuits. VHDL is a fairly general-purpose

language, although it requires a simulator on which to run the code. It can read and write files on the host computer, so a VHDL program can be written that generates another VHDL program to be integrated in the design being developed. The advantage of VHDL when used for systems design is that it allows the behavior of the required system to be described and simulated synthesis tools translate the design into real hardware.

# CHAPTER 3

# METHODOLOGY

## 3.1 Methodology Flow Chart

For this project, there were some steps or stage must be done to make it happen follow the path.

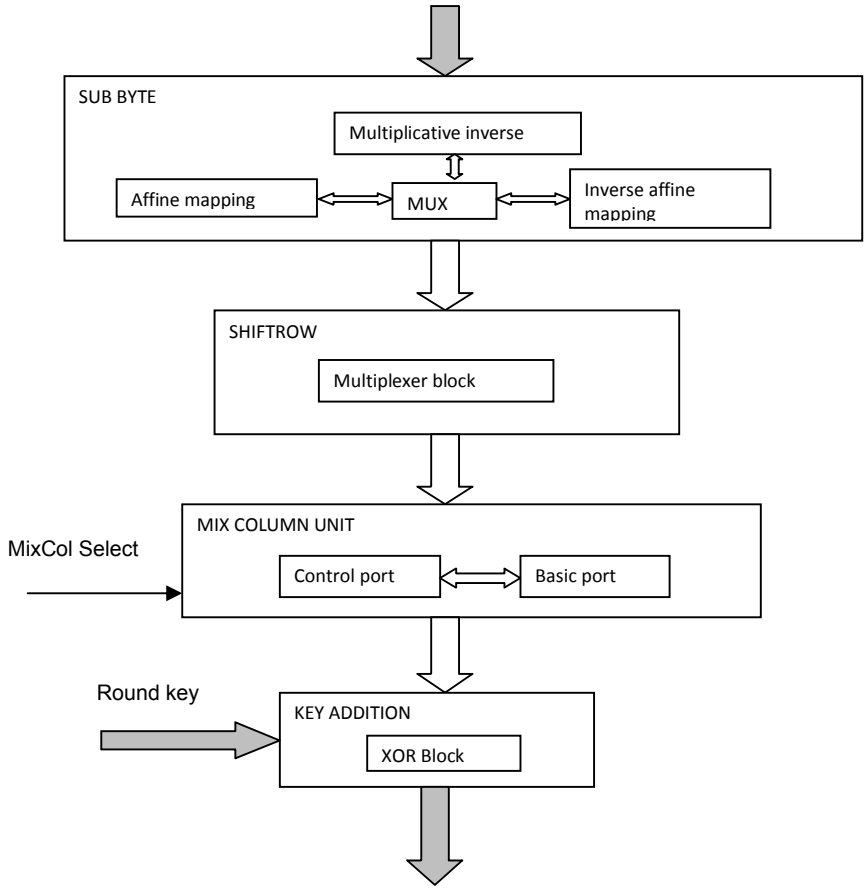


**Figure 3.1:** AES flow chart of encrypting data

### 3.1.1 SubBytes

SubBytes are a transformation in the cipher that processes the state using a non-linear substitution step where each byte independently replaced with another according to a lookup table (S-Box).

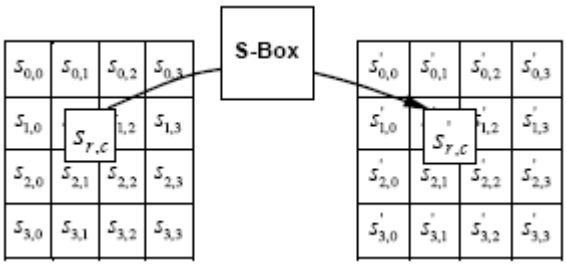


**Figure 3.2:** SubByte applies the S-Box to each byte of the state

The S-Box used in this transformation presented in hexadecimal form for the operation. For example, if $S_{1,1}$ = {79}, then the substitution value would be determined by intersection of the row '7' and the column '9' in Fig. 3.2. The result in $S'_{1,1}$ having a value {b6}.

| | | y | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | a | b | c | d | e | f |
| x | 0 | 63 | 7c | 77 | 7b | f2 | 6b | 6f | c5 | 30 | 01 | 67 | 2b | fe | d7 | ab | 76 |
| | 1 | ca | 82 | c9 | 7d | fa | 59 | 47 | f0 | ad | d4 | a2 | af | 9c | a4 | 72 | c0 |
| | 2 | b7 | fd | 93 | 26 | 36 | 3f | f7 | cc | 34 | a5 | e5 | f1 | 71 | d8 | 31 | 15 |
| | 3 | 04 | c7 | 23 | c3 | 18 | 96 | 05 | 9a | 07 | 12 | 80 | e2 | eb | 27 | b2 | 75 |
| | 4 | 09 | 83 | 2c | 1a | 1b | 6e | 5a | a0 | 52 | 3b | d6 | b3 | 29 | e3 | 2f | 84 |
| | 5 | 53 | d1 | 00 | ed | 20 | fc | b1 | 5b | 6a | cb | be | 39 | 4a | 4c | 58 | cf |
| | 6 | d0 | ef | aa | fb | 43 | 4d | 33 | 85 | 45 | f9 | 02 | 7f | 50 | 3c | 9f | a8 |
| | 7 | 51 | a3 | 40 | 8f | 92 | 9d | 38 | f5 | bc | b6 | da | 21 | 10 | ff | f3 | d2 |
| | 8 | cd | 0c | 13 | ec | 5f | 97 | 44 | 17 | c4 | a7 | 7e | 3d | 64 | 5d | 19 | 73 |
| | 9 | 60 | 81 | 4f | dc | 22 | 2a | 90 | 88 | 46 | ee | b8 | 14 | de | 5e | 0b | db |
| | a | e0 | 32 | 3a | 0a | 49 | 06 | 24 | 5c | c2 | d3 | ac | 62 | 91 | 95 | e4 | 79 |
| | b | e7 | c8 | 37 | 6d | 8d | d5 | 4e | a9 | 6c | 56 | f4 | ea | 65 | 7a | ae | 08 |
| | c | ba | 78 | 25 | 2e | 1c | a6 | b4 | c6 | e8 | dd | 74 | 1f | 4b | bd | 8b | 8a |
| | d | 70 | 3e | b5 | 66 | 48 | 03 | f6 | 0e | 61 | 35 | 57 | b9 | 86 | c1 | 1d | 9e |
| | e | e1 | f8 | 98 | 11 | 69 | d9 | 8e | 94 | 9b | 1e | 87 | e9 | ce | 55 | 28 | df |
| | f | 8c | a1 | 89 | 0d | bf | e6 | 42 | 68 | 41 | 99 | 2d | 0f | b0 | 54 | bb | 16 |

**Figure 3.3:** S-Box: substitution values for the byte xy (in hexadecimal format)